

CLAIMS

What is claimed is:

1. An apparatus comprising:
a plurality of physical buffers to be used by operations associated with computer program instructions;
virtualization logic to map the physical buffers to a plurality of virtual buffers and to prevent two or more operations that share the same physical buffer from interfering with each other when accessing the same physical buffer.
2. The apparatus of claim 1 wherein the virtualization logic includes logic to set a head buffer pointer (HBP) to point to a last physical buffer within the plurality of physical buffers.
3. The apparatus of claim 2 wherein the virtualization logic includes logic to increment the HBP if a buffer is de-allocated.
4. The apparatus of claim 3 wherein the virtualization logic includes physical buffer check (PBC) logic to check whether a virtual buffer index is less than or equal to the HBP.
5. The apparatus of claim 4 wherein the PBC logic is within a scheduler unit within a microprocessor.

6. The apparatus of claim 5 wherein a first operation is stored within the scheduler unit if the virtual buffer index is not less than or equal to the HBP.
7. The apparatus of claim 5 wherein a buffer is allocated to an operation only if the virtual buffer index is less than or equal to the HBP.
8. The apparatus of claim 6 wherein the first operation is a load operation and the virtual buffer index is a virtual load buffer index.
9. A method comprising:
 - initializing a head buffer pointer (HBP) to point to a last physical buffer in a buffer stack;
 - checking whether a whether a virtual buffer index is less than or equal to the HBP;
 - allowing an operation access to a buffer within the buffer stack if the virtual buffer index is less than or equal to HBP,
 - otherwise denying the operation access to the buffer.
10. The method of claim 9 further comprising de-allocating the buffer after the operation is retired.

11. The method of claim 10 further comprising incrementing the HBP after the operation is retired.
12. The method of claim 11 wherein other operations are allowed access to the buffer after the HBP is incremented.
13. The method of claim 12 wherein the operation is a load operation and the buffer is a load buffer.
14. The method of claim 12 wherein the operation is a store operation and the buffer is a store buffer.
15. A system comprising:
 - a memory to store an instruction comprising an operation;
 - a processor comprising virtualization logic to map a plurality of physical buffers to be used by the operation to a plurality of virtual buffers, the processor further comprising buffer access management logic to prevent two or more operations from interfering with each other if they are to access the same physical buffers.
16. The system of claim 15 wherein the virtualization logic includes logic to set a head buffer pointer (HBP) to point to a last physical buffer within the plurality of physical buffers.

17. The system of claim 16 wherein the virtualization logic includes logic to increment the HBP if a buffer is de-allocated.
18. The system of claim 17 wherein the virtualization logic includes physical buffer check (PBC) logic to check whether a virtual buffer index is less than or equal to the HBP.
19. The system of claim 18 wherein the PBC logic is within a scheduler unit within the processor.
20. The system of claim 19 wherein a first operation is stored within the scheduler unit if the virtual buffer index is not less than or equal to the HBP.
21. The system of claim 20 wherein a buffer is allocated to an operation only if the virtual buffer index is less than or equal to the HBP.
22. The system of claim 21 wherein the first operation is a load operation and the virtual buffer index is a virtual load buffer index.
23. The system of claim 21 wherein the first operation is a store operation and the virtual buffer index is a virtual store buffer index.

24. A machine-readable medium having stored thereon a set of instructions, which if executed by a machine cause the machine to perform a method comprising:
- initializing a head buffer pointer (HBP) to point to a last physical buffer in a buffer stack;
 - checking whether a whether a virtual buffer index is less than or equal to the HBP;
 - allowing an operation access to a buffer within the buffer stack if the virtual buffer index is less than or equal to HBP, otherwise denying the operation access to the buffer.
25. The machine-readable medium of claim 24 wherein the method further comprises de-allocating the buffer after the operation is retired.
26. The machine-readable medium of claim 25 wherein the method further comprises incrementing the HBP after the operation is retired.
27. The machine-readable medium of claim 26 wherein other operations are allowed access to the buffer after the HBP is incremented.
28. The machine-readable medium of claim 27 wherein the operation is a load operation and the buffer is a load buffer.

29. The machine-readable medium of claim 28 wherein the operation is a store operation and the buffer is a store buffer.